Refine Search

Search Results -

Terms	Documents
L2 same transfer$4	16

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search: L7

Refine Search

Recall Text Clear Interrupt

Search History

DATE: Tuesday, May 31, 2005 Printable Copy Create Case

Set Name side by side	Query	Hit Count	Set Name result set
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L7	l2 same transfer$4	16	L7
	DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR		
L6	l1 or L4	0	L6
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L5	l1 or L4	6	L5
L4	L2 same (transfer$4 near5 data)	6	L4
L3	L2 same (transfer$4 near3 data)	6	L3
L2	"state machine" same mode same "real time"	115	L2
L1	"state machine" same mode same "real time" same bridge	1	L1

END OF SEARCH HISTORY

Freeform Search

Database:
US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Term: l1 or L4

Display: 10 **Documents in Display Format:** - **Starting with Number** 1

Generate: ○ **Hit List** ● **Hit Count** ○ **Side by Side** ○ **Image**

Search Clear Interrupt

Search History

DATE: Tuesday, May 31, 2005 Printable Copy Create Case

Set Name side by side	**Query**	**Hit Count**	**Set Name** result set
	DB=PGPB,USPT,USOC; PLUR=YES; OP=OR		
L5	l1 or L4	6	L5
L4	L2 same (transfer$4 near5 data)	6	L4
L3	L2 same (transfer$4 near3 data)	6	L3
L2	"state machine" same mode same "real time"	115	L2
L1	"state machine" same mode same "real time" same bridge	1	L1

END OF SEARCH HISTORY

Refine Search

Search Results -

Terms	Documents
L1 or L4	0

Database:

US Pre-Grant Publication Full-Text Database
US Patents Full-Text Database
US OCR Full-Text Database
EPO Abstracts Database
JPO Abstracts Database
Derwent World Patents Index
IBM Technical Disclosure Bulletins

Search:

```
L6
```

Refine Search

Recall Text   Clear   Interrupt

Search History

DATE: Tuesday, May 31, 2005 Printable Copy Create Case

Set Name side by side	**Query**	**Hit Count**	**Set Name** result set
DB=EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=OR			
L6	l1 or L4	0	L6
DB=PGPB,USPT,USOC; PLUR=YES; OP=OR			
L5	l1 or L4	6	L5
L4	L2 same (transfer$4 near5 data)	6	L4
L3	L2 same (transfer$4 near3 data)	6	L3
L2	"state machine" same mode same "real time"	115	L2
L1	"state machine" same mode same "real time" same bridge	1	L1

END OF SEARCH HISTORY

Home | Login | Logout | Access Information | Alerts | Sitemap | Help

Welcome United States Patent and Trademark Office

Search Results BROWSE SEARCH IEEE XPLORE GUIDE SUPPORT

Results for "**(state machine<in>metadata) <and> (real time<in>metadata) and bridge**" e-mail printer friendly

Your search matched **7** of **1164322** documents.

A maximum of **100** results are displayed, **25** to a page, sorted by **Relevance** in **Descending** order.

» View Session History

» New Search

Modify Search

(state machine<in>metadata) <and> (real time<in>metadata) and bridge »

□ Check to search only within this results set

Display Format: ● Citation ○ Citation & Abstract

» **Key**

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

Select Article Information

□ 1. **Active objects: a paradigm for communications and event driven systems**
Caal, G.; Divin, A.; Petitpierre, C.;
Global Telecommunications Conference, 1994. GLOBECOM '94. 'Communications: The Global Bridge'., IEEE
28 Nov.-2 Dec. 1994 Page(s):485 - 489 vol.1
AbstractPlus | Full Text: PDF(408 KB) IEEE CNF

□ 2. **A PC-DSP-based unified control system design for FACTS devices**
Zhang, L.; Yang, Z.; Chen, S.; Crow, M.L.;
Power Engineering Society Winter Meeting, 2001. IEEE
Volume 1, 28 Jan.-1 Feb. 2001 Page(s):252 - 257 vol.1
AbstractPlus | Full Text: PDF(676 KB) IEEE CNF

□ 3. **A CMOS 510 K-transistor single-chip token-ring LAN controller (TRC) compatible with IEEE802.5 MAC protocol**
Kanuma, A.; Yaguchi, T.; Tanaka, K.; Katsumata, E.; Fujimoto, K.; Miyazawa, Y.; Iida, S.I.; Yamamoto, T.;
Solid-State Circuits, IEEE Journal of
Volume 25, Issue 1, Feb. 1990 Page(s):132 - 141
AbstractPlus | Full Text: PDF(812 KB) IEEE JNL

□ 4. **An object oriented Petri net language for embedded system design**
Esser, R.;
Software Technology and Engineering Practice, 1997. Proceedings., Eighth IEEE International Workshop on [incorporating Computer Aided Software Engineering]
14-18 July 1997 Page(s):216 - 223
AbstractPlus | Full Text: PDF(684 KB) IEEE CNF

□ 5. **Detection of response time failures of real-time software**
Pekilis, B.R.; Seviora, R.E.;
PROCEEDINGS The Eighth International Symposium On Software Reliability Engineering
2-5 Nov. 1997 Page(s):38 - 47
AbstractPlus | Full Text: PDF(896 KB) IEEE CNF

□ 6. **SpecTRM: a CAD system for digital automation**
Leveson, N.G.; Reese, J.D.; Heimdahl, M.P.E.;
Digital Avionics Systems Conference, 1998. Proceedings., 17th DASC. The AIAA/IEEE/SAE
Volume 1, 31 Oct.-7 Nov. 1998 Page(s):B52/1 - B52/8 vol.1
AbstractPlus | Full Text: PDF(720 KB) IEEE CNF

□ 7. **Modular code generation from hybrid automata based on data dependency**
Jesung Kim; Insup Lee;
Real-Time and Embedded Technology and Applications Symposium, 2003. Proceedings. The 9th IEEE
27-30 May 2003 Page(s):160 - 168

AbstractPlus | Full Text: PDF(435 KB) IEEE CNF

Access this document

Full Text: PDF (454 KB)

Download this citation

Choose Citation

Download EndNote,ProCite,RefMan

» Learn More

Rights & Permissions

Request Permissions RIGHTSLINK

» Learn More

Active objects: a paradigm for communications and event driven systems

Caal, G. Divin, A. Petitpierre, C.
Lab. de Teleinf., Ecole Polytech. Federale de Lausanne, Switzerland;

This paper appears in: **Global Telecommunications Conference, 1994. GLOBECOM '94. 'Communications: The Global Bridge'., IEEE**
Publication Date: 28 Nov.-2 Dec. 1994
On page(s): 485 - 489 vol.1
Meeting Date: 11/28/1994 - 12/02/1994
Location: San Francisco, CA
INSPEC Accession Number:5079786
DOI: 10.1109/GLOCOM.1994.513568
Posted online: 2002-08-06 19:17:55.0

Abstract

Current techniques for handling events, such as the "callback functions" approach, present some limitations to the development of network software, communication layered protocols, graphical interfaces for **real-time** applications and multimedia. These techniques lack the mechanisms for data flow structuring, for the interaction between different events and for finite **state machine** design. This paper presents a paradigm, based on the concept of active objects, that provides a new way of designing event driven applications. The concurrent and object-oriented capabilities of this approach make it particularly useful for facilitating the construction of complex event driven systems

Index Terms

Inspec

Controlled Indexing

C language C++ programming UNIX techniques Unix active objects callback functions communication layered protocols concurrent capabilities data flow structuring discrete event simulation event driven systems finite **state machine** design graphical interfaces graphical user interfaces interrupts multimedia multiplexing network software object-oriented capabilities object-oriented languages object-oriented programming protocols **real-time** applications telecommunication computing telecommunication networks

Non-controlled Indexing

C language C++ programming UNIX techniques Unix active objects callback functions communication layered protocols concurrent capabilities data flow structuring discrete event simulation event driven systems finite **state machine** design graphical interfaces graphical user interfaces interrupts multimedia multiplexing network software object-oriented capabilities object-oriented languages object-oriented programming protocols **real-time** applications telecommunication computing telecommunication networks

Author Keywords

Not Available

References

No references available on IEEE Xplore.

Citing Documents

No citing documents available on IEEE Xplore.

View Search Results | Next Article

Help Contact Us Privacy & Security IEEE.org

First Hit Previous Doc Next Doc Go to Doc#

End of Result Set

□ Generate Collection Print

L1: Entry 1 of 1 File: PGPB Mar 17, 2005

PGPUB-DOCUMENT-NUMBER: 20050060479
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20050060479 A1

TITLE: High speed and flexible control for bridge controllers

PUBLICATION-DATE: March 17, 2005

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Deng, Brian Tse	Richardson	TX	US	
Nie, Dinghui Richard	Plano	TX	US	
Erickson, Joseph M.	Frisco	TX	US	

APPL-NO: 10/ 651524 [PALM]
DATE FILED: August 29, 2003

INT-CL: [07] G06 F 13/36

US-CL-PUBLISHED: 710/306
US-CL-CURRENT: 710/306

REPRESENTATIVE-FIGURES: 2

ABSTRACT:

A bridge controller controls the data flow to/from a USB bus to/from an ATA/ATAPI drive, such as an ATA hard drive or ATAPI CD or DVD drive. The bridge controller has a state machine which receives the CBW in a background mode in real time as the packet is being transferred to the bridge controller. The state machine uses the CBW to set up the data transfer. The bridge controller also has a programmable processor which is coupled to the CBW once it is received in a buffer memory. The programmable processor makes changes in the set up of the receiving device for the transfer, if needed, and initiates the data transfer.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L5: Entry 1 of 6 File: PGPB Mar 17, 2005

DOCUMENT-IDENTIFIER: US 20050060479 A1
TITLE: High speed and flexible control for bridge controllers

Abstract Paragraph:
A bridge controller controls the data flow to/from a USB bus to/from an ATA/ATAPI drive, such as an ATA hard drive or ATAPI CD or DVD drive. The bridge controller has a state machine which receives the CBW in a background mode in real time as the packet is being transferred to the bridge controller. The state machine uses the CBW to set up the data transfer. The bridge controller also has a programmable processor which is coupled to the CBW once it is received in a buffer memory. The programmable processor makes changes in the set up of the receiving device for the transfer, if needed, and initiates the data transfer.

Summary of Invention Paragraph:
[0007] This and other objects and features of the invention are provided, in accordance with one aspect of the invention, by a bridge controller for transferring data between a data storage device and a data utilization device, the bridge controller receiving a command information packet for controlling the data transfer. A state machine receives command information in a background mode in real time as the packet is being transferred to the bridge controller, the state machine utilizing the command information to set up the receiving device for the data transfer. A programmable processor is coupled to the command information packet after the packet has been received, the processor making changes to the set up of the receiving device for the transfer, if needed, and then initiating the data transfer.

Summary of Invention Paragraph:
[0008] Another aspect of the invention includes a USB to ATA/ATAPI bridge. A physical layer receives serial command data from the USB bus and converts the data to a parallel format. A transfer controller receives the parallel data and transfers the data to a buffer memory. A state machine operating in background mode on the parallel data flowing through the transfer controller in real time sets up the ATA or ATAPI device for a data transfer. A programmable processor is coupled to the buffer memory and being interrupted after all command information has been received, to individually alter any set up data for the ATA or ATAPI device that is needed, and then initiates the data transfer.

Summary of Invention Paragraph:
[0009] A third aspect of the invention comprises a method of operating a USB to ATA or ATAPI bridge. Command data is transferred from a data utilization device via a USB bus through a data transfer device to a buffer memory. A state machine is operated in a background mode using data flowing through the data transfer device in real time to extract set up data and store the data in the required command-related registers to set up a data transfer. A programmable processor utilizes the data stored in the buffer memory to individually alter the command-related data for the ATA or ATAPI device that is needed. The data transfer is then initiated.

Detail Description Paragraph:
[0019] The first state in the state machine is the idle state labeled "CBW_IDLE". This state is an idle state waiting for the output data packet address to this node and the acquisition of data in real time as it is being transferred to the bridge controller takes place. If this data acquisition mode is enabled by the signal, labeled the "snoop" CBW enable in block 301, the state machine receives the first data packet and looks at the first data quadlet in block 302 to see if it matches the dCBWSignature. In this example, the signature would be "0x43425355" which is the ASCII code "CBSU" which means a USB mass storage class command. If this first data quadlet matches the signature, the machine goes to the state "WAIT_TAG". If the data does not match the signature, the state machine will ignore this bit packet and go to the state "WAIT_EOT" to wait for end of the transaction at block 354.

CLAIMS:

1. A bridge controller for transferring data between a data storage device and a data utilization device, the bridge controller receiving a command information packet for controlling the data transfer, comprising: a state machine receiving command information in a background mode in real time as the packet is being transferred to the bridge controller, the state machine utilizing the command information to set up the receiving device for the data transfer; and a programmable processor coupled to the command information packet after the packet has been received, the processor making changes to the set up of the receiving device for the transfer, if needed, and then initiating the data transfer.

11. A USB to ATA/ATAPI bridge comprising: a physical layer receiving serial command data from the USB bus and converting the data to a parallel format; a transfer controller receiving the parallel data for transferring the data to a buffer memory; a state machine operating in background mode on the parallel data flowing through the transfer controller in real time to set up the ATA or ATAPI device for a data transfer; and a programmable processor coupled to the buffer memory and being interrupted after all command information has been received, to individually alter any set up data for the ATA or ATAPI device that is needed, and then initiating the data transfer.

19. A method of operating a USB to ATA or ATAPI bridge comprising: transferring command data from a data utilization device via a USB bus through a data transfer device to a buffer memory; operating a state machine in a background mode using data flowing through the data transfer device in real time to extract set up data and store the data to set up a data transfer; operating a programmable processor utilizing the data stored in the buffer memory to individually alter the command-related data for the ATA or ATAPI device that is needed; and initiating the data transfer.

Previous Doc Next Doc Go to Doc#

Access this document

Full Text: PDF (454 KB)

Download this citation

Choose Citation

Download EndNote,ProCite,RefMan

» Learn More

Rights & Permissions

Request Permissions RIGHTSLINK

» Learn More

Active objects: a paradigm for communications and event driven systems

Caal, G. Divin, A. Petitpierre, C.
Lab. de Teleinf., Ecole Polytech. Federale de Lausanne, Switzerland;

This paper appears in: **Global Telecommunications Conference, 1994. GLOBECOM '94. 'Communications: The Global Bridge'., IEEE**
Publication Date: 28 Nov.-2 Dec. 1994
On page(s): 485 - 489 vol.1
Meeting Date: 11/28/1994 - 12/02/1994
Location: San Francisco, CA
INSPEC Accession Number:5079786
DOI: 10.1109/GLOCOM.1994.513568
Posted online: 2002-08-06 19:17:55.0

Abstract

Current techniques for handling events, such as the "callback functions" approach, present some limitations to the development of network software, communication layered protocols, graphical interfaces for **real-time** applications and multimedia. These techniques lack the mechanisms for data flow structuring, for the interaction between different events and for finite **state machine** design. This paper presents a paradigm, based on the concept of active objects, that provides a new way of designing event driven applications. The concurrent and object-oriented capabilities of this approach make it particularly useful for facilitating the construction of complex event driven systems

Index Terms

Inspec

Controlled Indexing

C language C++ programming UNIX techniques Unix active objects callback functions communication layered protocols concurrent capabilities data flow structuring discrete event simulation event driven systems finite **state machine** design graphical interfaces graphical user interfaces interrupts multimedia multiplexing network software object-oriented capabilities object-oriented languages object-oriented programming protocols **real-time** applications telecommunication computing telecommunication networks

Non-controlled Indexing

C language C++ programming UNIX techniques Unix active objects callback functions communication layered protocols concurrent capabilities data flow structuring discrete event simulation event driven systems finite **state machine** design graphical interfaces graphical user interfaces interrupts multimedia multiplexing network software object-oriented capabilities object-oriented languages object-oriented programming protocols **real-time** applications telecommunication computing telecommunication networks

Author Keywords
Not Available

References

No references available on IEEE Xplore.

Citing Documents

No citing documents available on IEEE Xplore.

◂ View Search Results | Next Article ▸

Help Contact Us Privacy & Security IEEE.org

Indexed by
Inspec®

Hit List

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Search Results - Record(s) 1 through 6 of 6 returned.

□ 1. Document ID: US 20050060479 A1

L5: Entry 1 of 6 File: PGPB Mar 17, 2005

PGPUB-DOCUMENT-NUMBER: 20050060479
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20050060479 A1

TITLE: High speed and flexible control for bridge controllers

PUBLICATION-DATE: March 17, 2005

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Deng, Brian Tse	Richardson	TX	US	
Nie, Dinghui Richard	Plano	TX	US	
Erickson, Joseph M.	Frisco	TX	US	

US-CL-CURRENT: 710/306

Full | Title | Citation | Front | Review | Classification | Date | Reference | Sequences | Attachments | Claims | KWIC | Draw Desc | Image

□ 2. Document ID: US 6564339 B1

L5: Entry 2 of 6 File: USPT May 13, 2003

US-PAT-NO: 6564339
DOCUMENT-IDENTIFIER: US 6564339 B1

TITLE: Emulation suspension mode handling multiple stops and starts

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw Desc | Image

□ 3. Document ID: US 6553513 B1

L5: Entry 3 of 6 File: USPT Apr 22, 2003

US-PAT-NO: 6553513
DOCUMENT-IDENTIFIER: US 6553513 B1

TITLE: Emulation suspend mode with differing response to differing classes of interrupts

Full | Title | Citation | Front | Review | Classification | Date | Reference | Claims | KWIC | Draw Desc | Image

□ 4. Document ID: US 6081885 A

L5: Entry 4 of 6 File: USPT Jun 27, 2000

US-PAT-NO: 6081885
DOCUMENT-IDENTIFIER: US 6081885 A

TITLE: Method and apparatus for halting a processor and providing state visibility on a pipeline phase basis

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw. Desc | Image

□ 5. Document ID: US 6065106 A

L5: Entry 5 of 6 File: USPT May 16, 2000

US-PAT-NO: 6065106
DOCUMENT-IDENTIFIER: US 6065106 A

TITLE: Resuming normal execution by restoring without refetching instructions in multi-word instruction register interrupted by debug instructions loading and processing

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw. Desc | Image

□ 6. Document ID: US 5970241 A

L5: Entry 6 of 6 File: USPT Oct 19, 1999

US-PAT-NO: 5970241
DOCUMENT-IDENTIFIER: US 5970241 A

TITLE: Maintaining synchronism between a processor pipeline and subsystem pipelines during debugging of a data processing system

Full | Title | Citation | Front | Review | Classification | Date | Reference | | | Claims | KWIC | Draw. Desc | Image

Clear | Generate Collection | Print | Fwd Refs | Bkwd Refs | Generate OACS

Terms	Documents
L1 or L4	6

Display Format: - Change Format

Previous Page Next Page Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L7: Entry 2 of 16 File: PGPB Feb 19, 2004

PGPUB-DOCUMENT-NUMBER: 20040034822
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20040034822 A1

TITLE: Implementing a scalable, dynamic, fault-tolerant, multicast based file transfer and asynchronous file replication protocol

PUBLICATION-DATE: February 19, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Marchand, Benoit	Montreal		CA	

APPL-NO: 10/ 445145 [PALM]
DATE FILED: May 23, 2003

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	DOC-ID	APPL-DATE
EP	EP 02011310.6	2002EP-EP 02011310.6	May 23, 2002

INT-CL: [07] G01 R 31/28

US-CL-PUBLISHED: 714/712
US-CL-CURRENT: 714/712

REPRESENTATIVE-FIGURES: 2

ABSTRACT:

Apparatus and method to improve the speed, scalability, robustness and dynamism of multicast data transfers to remote computers. Many Grid Computing applications, such as Genomics, Proteomics, Seismic, Risk Management, etc., require a priori transfer of sets of files or other data to remote computers prior to processing taking place. Existing multicast and data transfer protocols are static and can not guarantee that all nodes will contain a copy of the replicated data or files. The fully distributed data transfer and data replication protocol of the invention permits transfers which minimize processing requirements on master transfer nodes by spreading work across the network. The result is higher scalability than current centralized protocols, more dynamism and allows fault-tolerance by distribution of functionality. The ability to distribute the protocol is simplified through our innovative symmetric-connectionless data transfer protocol.

Previous Doc Next Doc Go to Doc#

First Hit Fwd Refs Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L7: Entry 11 of 16 File: USPT Feb 15, 2000

DOCUMENT-IDENTIFIER: US 6026467 A
TITLE: Content-addressable memory implemented with a memory management unit

Brief Summary Text (5):
A CAM makes it possible to handle list searches and data translation as embedded functions within a system. The combination of a CAM and a state machine creates an economical controller for real-time processes that need to perform look-ups, data translations, and entry maintenance in sparsely populated tables--ones with few entries compared to the address space required for direct table look-up. For example, an asynchronous transfer mode (ATM) switch must search internal tables that hold the necessary information for each connection that routes through the switch. The index to these tables is the virtual-path identifier (VPI) for the VPI/virtual channel identifier (VCI) combination from the header of an incoming data cell. The switch uses this information to look up the VPI and VCI for the outgoing link, the internal path through the switch to the correct output port, billing rates, traffic-flow parameters, flags for any special functions, etc. A CAM is particularly suited for such an application.

Previous Doc Next Doc Go to Doc#

L7: Entry 9 of 16 File: USPT May 16, 2000

DOCUMENT-IDENTIFIER: US 6065106 A
TITLE: Resuming normal execution by restoring without refetching instructions in multi-word instruction register interrupted by debug instructions loading and processing

Detailed Description Paragraph Table (74):
TABLE 45 ______ MTAP Status ______ STRAP Strap Device Mode. This bit is set from the ECR's device mode bits. When set it indicates the megamodule is in strap mode. This bit will be used to determine if the mode is switched unexpectedly (from emulation or test to strap). TCLK.sub.-- SEL Test Clock Selected. This bit, when one, indicates the test clock (TCLK) has been switched to by all unlocked domains. When TCLK is enabled the unlocked Domains are ready to perform data scans at the TLCK rate (scan clock switching is described later) and/or are running their functional clock at the TCLK rate. The state of the TCLK.sub.-- SEL status bit is latched and not modified until the clock switch is complete (SWINPROG inactive). IRBUSY Instruction Register Busy. This bit indicates the JTAG interface is busy with another operation (see section 10.2.4 for more information). Since the JTAG IR is shadowed the IR can be scanned (SHIFT.sub.-- IR) but cannot be updated (UPDATE.sub.-- IR) until the IRBUSY bit is clear. This bit is in the logic zero state after the JTAG TLR state and will remain a zero until a MTAP command is issued. SYNC.sub.-- ERR Sync Scan Error. This bit indicates that a data scan was attempted when MTAP 305 module was in a state that the scan could not be started. The scan is aborted and clocks not switched. This bit blocks both scan and CSM stimulus. This status bit is cleared by locking the ECR DONE.sub.-- TGLE Done Toggle. This status bit indicates CPU.sub.-- DONE has gone from inactive to active since the last IR Stop Status scan. The rising edge of the CPU.sub.-- DONE signal is captured by the DONE.sub.-- TGLE SRL. If Stop Status is selected the DONE.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and cleared. RST.sub.-- TGLE Reset Toggle. This status bit indicates RST.sub.-- TKN has gone from inactive to active since the last IR Error Status scan. The rising edge of the RST.sub.-- TKN signal is captured by the RST.sub.-- TGLE SRL. If Error Status is selected the RST.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the RST.sub.-- TGLE SRL is cleared. MINT.sub.-- TGLE Message Interrupt Toggle. This status bit indicates MINT.sub.-- TKN has gone from inactive to active since the last IR Real Time Status scan. The rising edge of the MINT.sub.-- TKN signal is captured by the MINT.sub.-- TGLE SRL. If Real Time Status is selected the MINT.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the MINT.sub.-- TGLE SRL is cleared. AINT.sub.-- TGLE Analysis Interrupt Toggle. This status bit indicates AINT.sub.-- TKN has gone from inactive to active since the last IR Real Time Status scan. The rising edge of the AINT.sub.-- TKN signal is captured by the AINT.sub.-- TGLE SRL. If Real Time Status is selected the AINT.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the AINT.sub.-- TGLE SRL is cleared. MSGSW.sub.-- TGLE Message Switch Toggle. This status bit indicates MSGSW has gone from inactive to active since the last IR Real Time Status scan. The rising edge of the MSGSE signal is captured by the MSGSW.sub.-- TGLE SRL. If Real Time Status is selected the MSGSW.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the MSGSW.sub.-- TGLE SRL is cleared. STRY.sub.-- TGLE Stream Ready Error Toggle. The STRY.sub.-- TGLE SRL may only be set if the previous JTAG command was the SDAT.sub.-- STRM command. The STRY.sub.-- TGLE SRL is set if CPU.sub.-- DONE is not active on the rising edge of MTAP Counter's XFER.sub.-- DATA signal. This condition indicates the previous ld/st did not advance prior to the next data transfer to or from the CPU's EDDATA1 register. If Emulation Error Status is selected the STRY.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the STRY.sub.-- TGLE SRL is cleared. STSW.sub.-- TGLE Stream Switch Error Toggle. The STSW.sub.-- TGLE SRL may only be set if the previous JTAG command was the SDAT.sub.-- STRM command. The STSW.sub.-- TGLE SRL is set if the MPSD code in the ECR's TERM field is not driving the MPSD bus on the rising edge of MTAP Counter's

XFER.sub,-- DATA signal. This condition indicates that UCLK is running two slow in relationship to TCLK to support the data streaming function. If Emulation Error Status is selected the STSW.sub.-- TGLE signal is loaded into the IR shift register during the CAPTURE.sub.-- IR JTAG state and the STSW.sub.-- TGLE SRL is cleared. CSM.sub.-- EXE CSM EXE State. Code state machine EXE state bit (1 if EXE state applied to output of CSM). See Figure 37 for more information. CSM.sub.-- LOCK CSM Lock State. Code state machine LOCK state bit. See Figure 37 for more information. C1, C0, Ce CSM Output. This field is the CSM MPSD code applied to the DTPs. See Figure 37 for more information.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L7: Entry 2 of 16 File: PGPB Feb 19, 2004

DOCUMENT-IDENTIFIER: US 20040034822 A1
TITLE: Implementing a scalable, dynamic, fault-tolerant, multicast based file transfer and asynchronous file replication protocol

Detail Description Paragraph:
[0050] FIGS. 4 through 10 show the finite state machines used to implement the multicast/broadcast file transfer and file replication protocols for the user interface, file transfer master and file transfer slave processes and their related sub-processes. The mode of operation can allow multiple concurrent multicast/broadcast file transfers and overlapping of multicast/broadcast file transfer, transfer error recovery and file replication phases. Fault-tolerance, scalability and dynamism are achieved through real-time peer selection and communication persistence.

Previous Doc Next Doc Go to Doc#

First Hit Previous Doc Next Doc Go to Doc#

□ Generate Collection Print

L7: Entry 2 of 16 File: PGPB Feb 19, 2004

PGPUB-DOCUMENT-NUMBER: 20040034822
PGPUB-FILING-TYPE: new
DOCUMENT-IDENTIFIER: US 20040034822 A1

TITLE: Implementing a scalable, dynamic, fault-tolerant, multicast based file transfer and asynchronous file replication protocol

PUBLICATION-DATE: February 19, 2004

INVENTOR-INFORMATION:

NAME	CITY	STATE	COUNTRY	RULE-47
Marchand, Benoit	Montreal		CA	

APPL-NO: 10/ 445145 [PALM]
DATE FILED: May 23, 2003

FOREIGN-APPL-PRIORITY-DATA:

COUNTRY	APPL-NO	DOC-ID	APPL-DATE
EP	EP 02011310.6	2002EP-EP 02011310.6	May 23, 2002

INT-CL: [07] G01 R 31/28

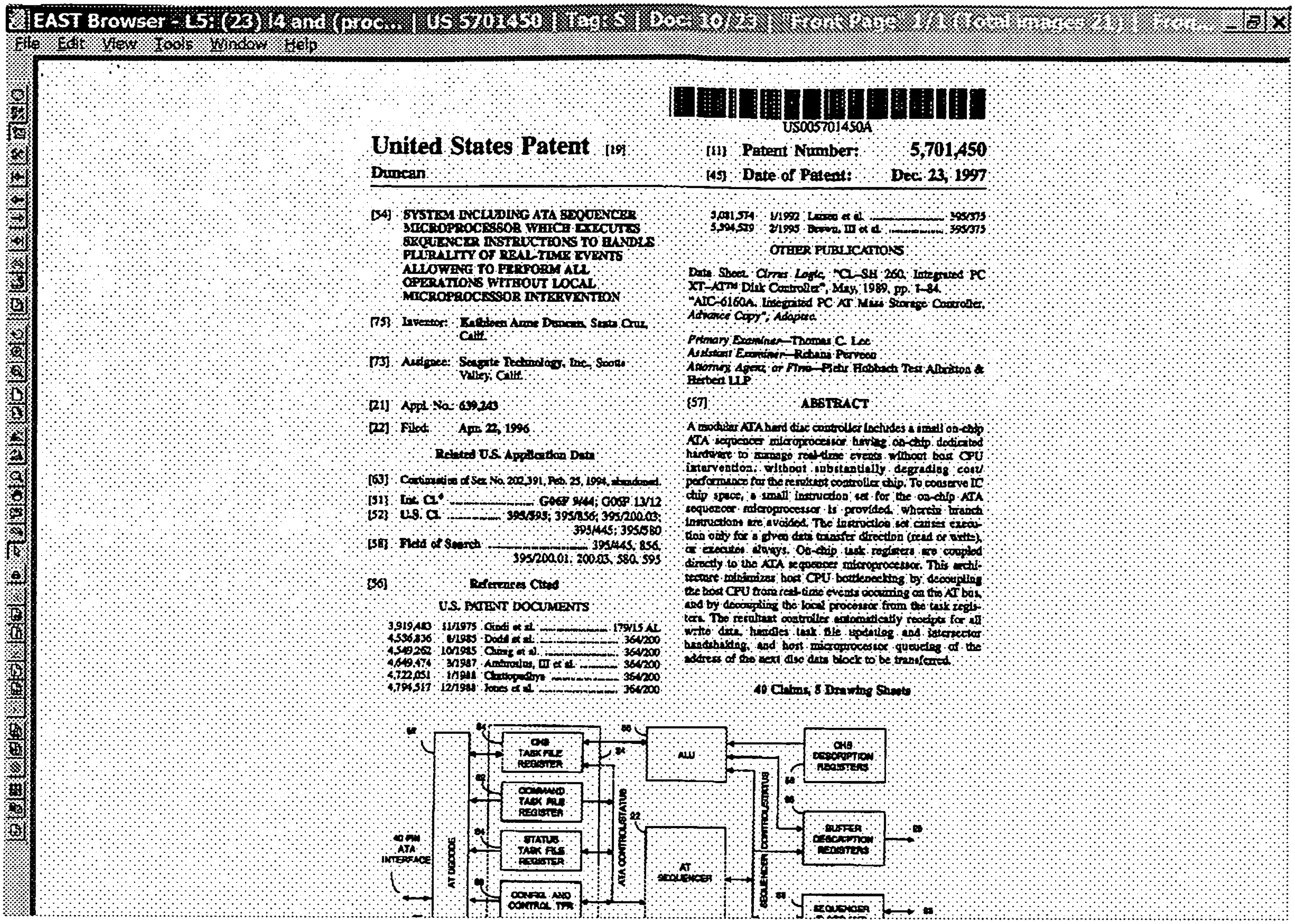
US-CL-PUBLISHED: 714/712
US-CL-CURRENT: 714/712

REPRESENTATIVE-FIGURES: 2

ABSTRACT:

Apparatus and method to improve the speed, scalability, robustness and dynamism of multicast data transfers to remote computers. Many Grid Computing applications, such as Genomics, Proteomics, Seismic, Risk Management, etc., require a priori transfer of sets of files or other data to remote computers prior to processing taking place. Existing multicast and data transfer protocols are static and can not guarantee that all nodes will contain a copy of the replicated data or files. The fully distributed data transfer and data replication protocol of the invention permits transfers which minimize processing requirements on master transfer nodes by spreading work across the network. The result is higher scalability than current centralized protocols, more dynamism and allows fault-tolerance by distribution of functionality. The ability to distribute the protocol is simplified through our innovative symmetric-connectionless data transfer protocol.

Previous Doc Next Doc Go to Doc#


EAST Browser - L5: (23) l4 and (proc... | US 5701450 | Tag: S | Doc: 10/23 | "Front Page" 1/1 (Total images 21) | Fron...
File
Edit
View
Tools
Window
Help
62
84
CHS TASK FILE REGISTER
82
COMMAND TASK FILE REGISTER
84
STATUS TASK FILE REGISTER
86
CONFIG. AND CONTROL TFR
40 PIN ATA INTERFACE
AT DECODE
ATA CONTROL/STATUS
94
58
ALU
22
AT SEQUENCER
SEQUENCER CONTROL/STATUS
CHS DESCRIPTION REGISTERS
58
86
BUFFER DESCRIPTION REGISTERS
88
58
SEQUENCER
88

US005701450A

United States Patent [19]

Duncan

[11] Patent Number: 5,701,450

[45] Date of Patent: Dec. 23, 1997

[54] SYSTEM INCLUDING ATA SEQUENCER MICROPROCESSOR WHICH EXECUTES SEQUENCER INSTRUCTIONS TO HANDLE PLURALITY OF REAL-TIME EVENTS ALLOWING TO PERFORM ALL OPERATIONS WITHOUT LOCAL MICROPROCESSOR INTERVENTION

[75] Inventor: Kathleen Anne Duncan, Santa Cruz, Calif.

[73] Assignee: Seagate Technology, Inc., Scotts Valley, Calif.

[21] Appl. No.: 639,243

[22] Filed: Apr. 22, 1996

Related U.S. Application Data

[63] Continuation of Ser. No. 202,391, Feb. 25, 1994, abandoned.

[51] Int. Cl.^{6} G06F 9/44; G06F 13/12

[52] U.S. Cl. 395/595; 395/856; 395/200.03; 395/445; 395/580

[58] Field of Search 395/445, 856, 395/200.01, 200.03, 580, 595

[56] References Cited

U.S. PATENT DOCUMENTS

3,919,483	11/1975	Gindi et al.	179/15 AL
4,536,836	8/1985	Dodd et al.	364/200
4,549,262	10/1985	Chung et al.	364/200
4,649,474	3/1987	Ambrosius, III et al.	364/200
4,722,051	1/1988	Chattopadhya	364/200
4,794,517	12/1988	Jones et al.	364/200
5,081,574	1/1992	Larsen et al.	395/375
5,394,529	2/1995	Brown, III et al.	395/375

OTHER PUBLICATIONS

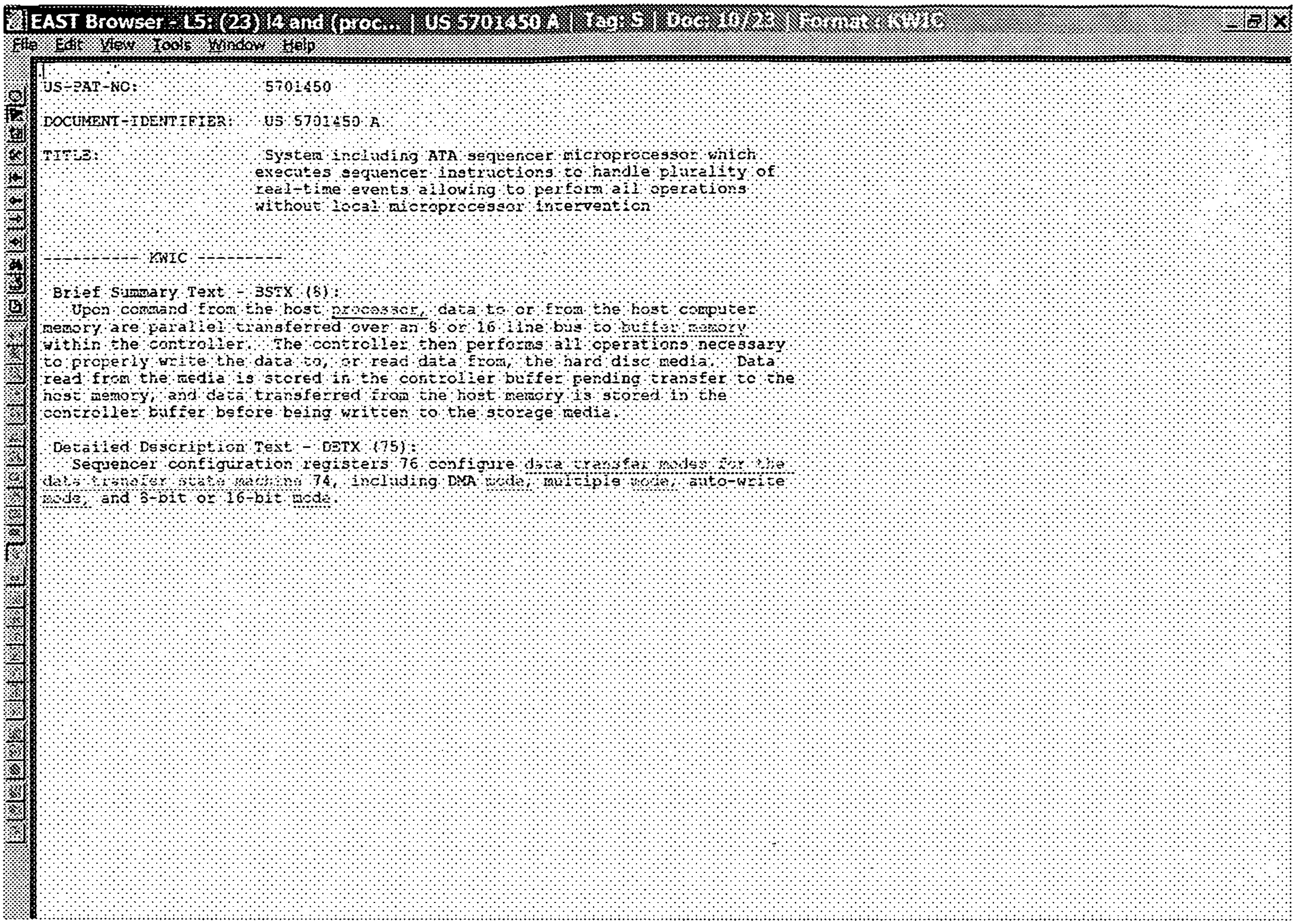
Data Sheet, *Cirrus Logic*, "CL-SH 260, Integrated PC XT-AT™ Disk Controller", May, 1989, pp. 1-84.
"AIC-6160A, Integrated PC AT Mass Storage Controller, *Advance Copy*", *Adaptec*.

Primary Examiner—Thomas C. Lee
Assistant Examiner—Rehana Perveen
Attorney, Agent, or Firm—Flehr Hohbach Test Albritton & Herbert LLP

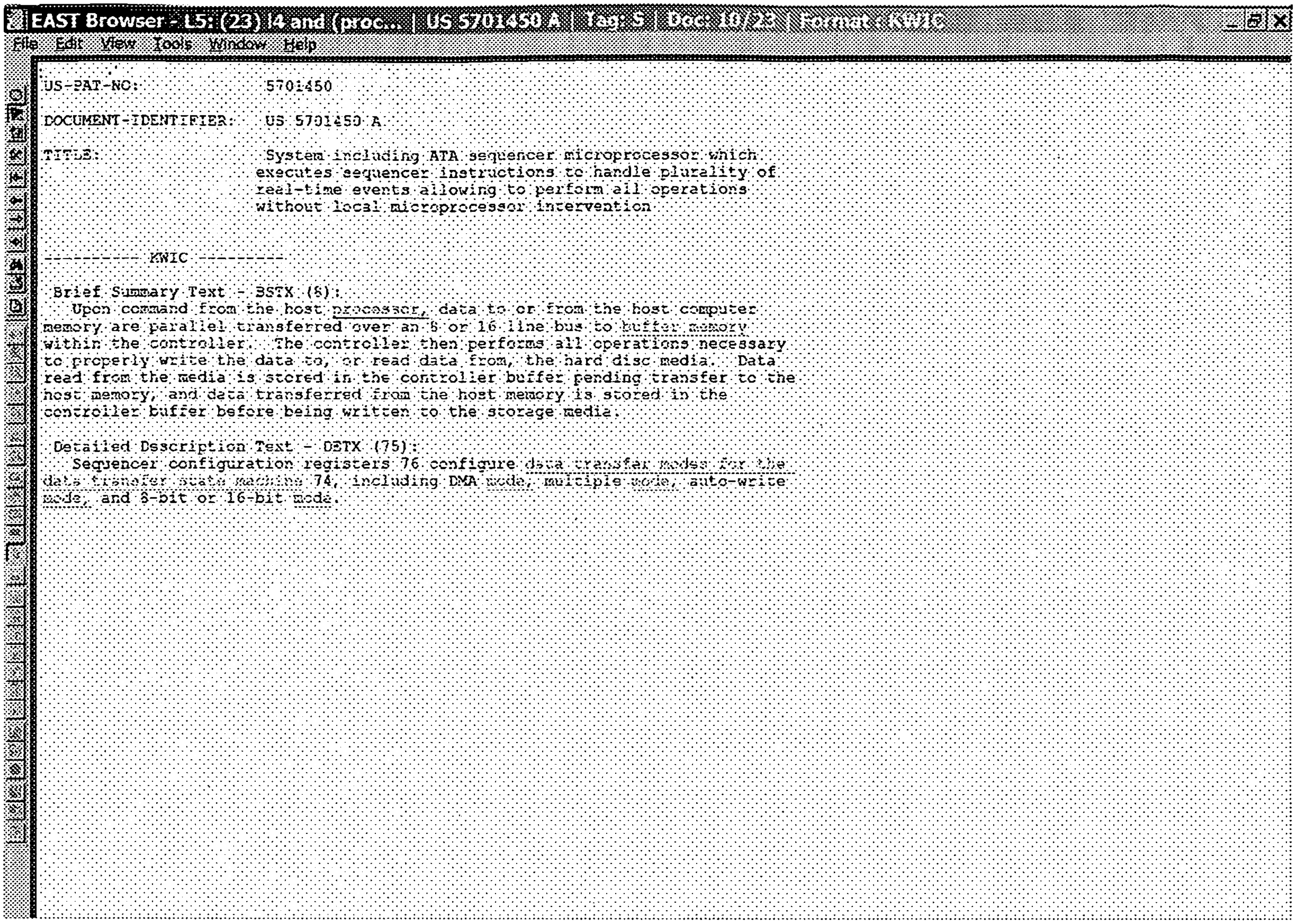
[57] ABSTRACT

A modular ATA hard disc controller includes a small on-chip ATA sequencer microprocessor having on-chip dedicated hardware to manage real-time events without host CPU intervention, without substantially degrading cost/performance for the resultant controller chip. To conserve IC chip space, a small instruction set for the on-chip ATA sequencer microprocessor is provided, wherein branch instructions are avoided. The instruction set causes execution only for a given data transfer direction (read or write), or executes always. On-chip task registers are coupled directly to the ATA sequencer microprocessor. This architecture minimizes host CPU bottlenecking by decoupling the host CPU from real-time events occurring on the AT bus, and by decoupling the local processor from the task registers. The resultant controller automatically receipts for all write data, handles task file updating and intersector handshaking, and host microprocessor queueing of the address of the next disc data block to be transferred.

40 Claims, 5 Drawing Sheets


EAST Browser - L5: (23) l4 and (proc... | US 5701450 A | Tag: S | Doc: 10/23 | Format : KWIC
File Edit View Tools Window Help

```
US-PAT-NO:                    5701450

DOCUMENT-IDENTIFIER:    US 5701450 A

TITLE:                         System including ATA sequencer microprocessor which
                               executes sequencer instructions to handle plurality of
                               real-time events allowing to perform all operations
                               without local microprocessor intervention


---------- KWIC ---------

 Brief Summary Text - BSTX (8):
   Upon command from the host processor, data to or from the host computer
memory are parallel transferred over an 8 or 16 line bus to buffer memory
within the controller.  The controller then performs all operations necessary
to properly write the data to, or read data from, the hard disc media.  Data
read from the media is stored in the controller buffer pending transfer to the
host memory, and data transferred from the host memory is stored in the
controller buffer before being written to the storage media.

 Detailed Description Text - DETX (75):
   Sequencer configuration registers 76 configure data transfer modes for the
data transfer state machine 74, including DMA mode, multiple mode, auto-write
mode, and 8-bit or 16-bit mode.
```


EAST Browser - L5: (23) l4 and (proc... | US 5701450 A | Tag: S | Doc: 10/23 | Format : KWIC
File Edit View Tools Window Help

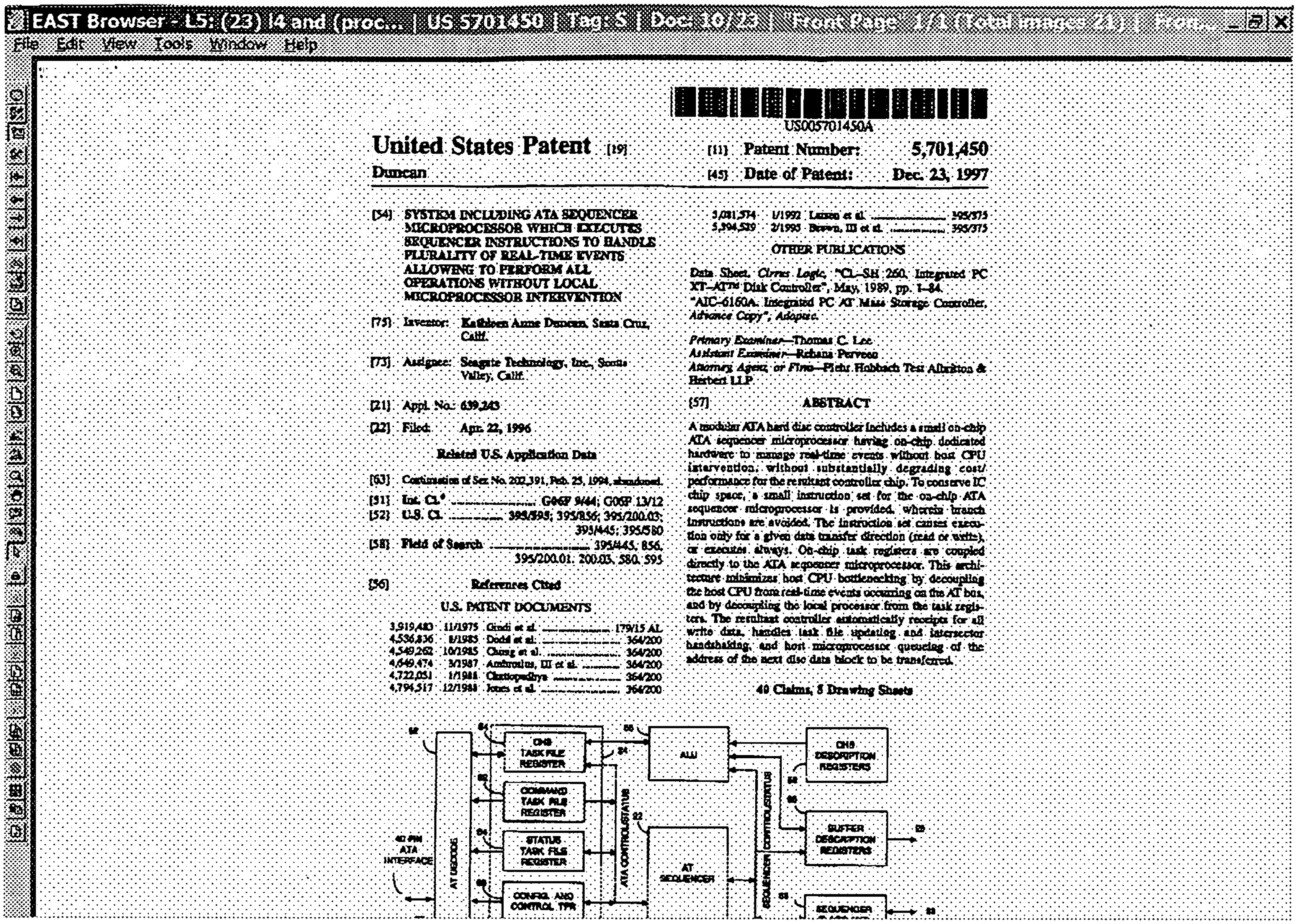
```
US-PAT-NO:                5701450

DOCUMENT-IDENTIFIER:      US 5701450 A

TITLE:                    System including ATA sequencer microprocessor which
                         executes sequencer instructions to handle plurality of
                         real-time events allowing to perform all operations
                         without local microprocessor intervention


---------- KWIC ---------
```

Brief Summary Text - BSTX (8):

Upon command from the host processor, data to or from the host computer memory are parallel transferred over an 8 or 16 line bus to buffer memory within the controller. The controller then performs all operations necessary to properly write the data to, or read data from, the hard disc media. Data read from the media is stored in the controller buffer pending transfer to the host memory, and data transferred from the host memory is stored in the controller buffer before being written to the storage media.

Detailed Dsscription Text - DETX (75):

Sequencer configuration registers 76 configure data transfer modes for the data transfer state machine 74, including DMA mode, multiple mode, auto-write mode, and 8-bit or 16-bit mode.


EAST Browser - L5: (23) l4 and (proc... | US 5701450 | Tag: S | Doc: 10/23 | "Front Page" 1/1 (Total images: 21) | From...
File Edit View Tools Window Help

US005701450A

United States Patent [19]

Duncan

[11] Patent Number: 5,701,450

[45] Date of Patent: Dec. 23, 1997

[54] SYSTEM INCLUDING ATA SEQUENCER MICROPROCESSOR WHICH EXECUTES SEQUENCER INSTRUCTIONS TO HANDLE PLURALITY OF REAL-TIME EVENTS ALLOWING TO PERFORM ALL OPERATIONS WITHOUT LOCAL MICROPROCESSOR INTERVENTION

[75] Inventor: Kathleen Anne Duncan, Santa Cruz, Calif.

[73] Assignee: Seagate Technology, Inc., Scotts Valley, Calif.

[21] Appl. No.: 639,243

[22] Filed: Apr. 22, 1996

Related U.S. Application Data

[63] Continuation of Ser. No. 202,391, Feb. 25, 1994, abandoned.

[51] Int. Cl.⁶ G06F 9/44; G06F 13/12

[52] U.S. Cl. 395/595; 395/856; 395/200.03; 395/445; 395/580

[58] Field of Search 395/445, 856, 395/200.01, 200.03, 580, 595

[56] References Cited

U.S. PATENT DOCUMENTS

3,919,483	11/1975	Gindi et al.	179/15 AL
4,536,836	8/1985	Dodd et al.	364/200
4,549,262	10/1985	Chung et al.	364/200
4,649,474	3/1987	Ambrosius, III et al.	364/200
4,722,051	1/1988	Chattopadhya	364/200
4,794,517	12/1988	Jones et al.	364/200
5,081,574	1/1992	Larsen et al.	395/375
5,394,529	2/1995	Brown, III et al.	395/375

OTHER PUBLICATIONS

Data Sheet, *Cirrus Logic*, "CL-SH 260, Integrated PC XT-AT™ Disk Controller", May, 1989, pp. 1-84.

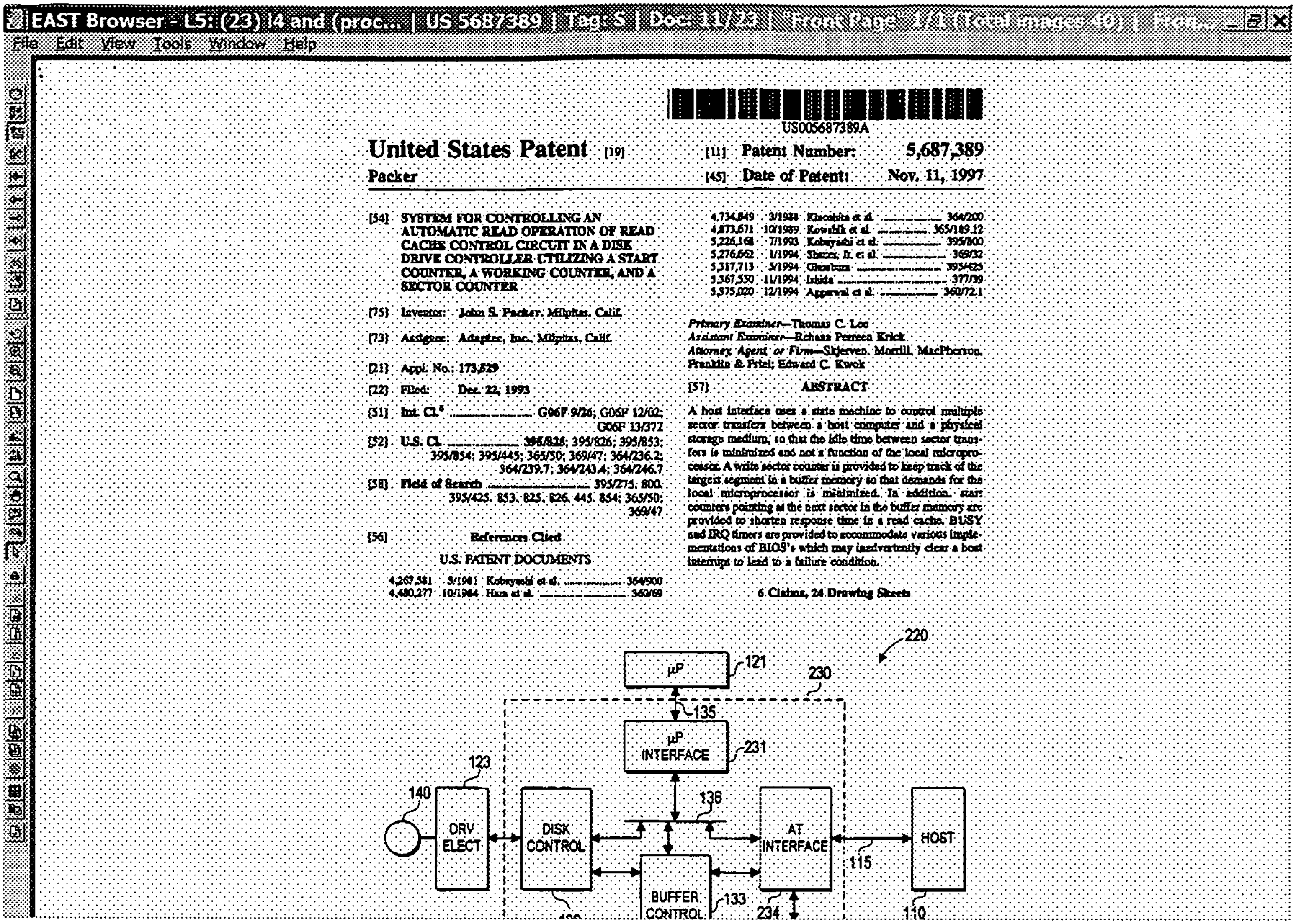
"AIC-6160A, Integrated PC AT Mass Storage Controller, *Advance Copy*", *Adaptec*.

Primary Examiner—Thomas C. Lee
Assistant Examiner—Rehana Perveen
Attorney, Agent, or Firm—Flehr Hohbach Test Albritton & Herbert LLP

[57] ABSTRACT

A modular ATA hard disc controller includes a small on-chip ATA sequencer microprocessor having on-chip dedicated hardware to manage real-time events without host CPU intervention, without substantially degrading cost/performance for the resultant controller chip. To conserve IC chip space, a small instruction set for the on-chip ATA sequencer microprocessor is provided, wherein branch instructions are avoided. The instruction set causes execution only for a given data transfer direction (read or write), or executes always. On-chip task registers are coupled directly to the ATA sequencer microprocessor. This architecture minimizes host CPU bottlenecking by decoupling the host CPU from real-time events occurring on the AT bus, and by decoupling the local processor from the task registers. The resultant controller automatically receipts for all write data, handles task file updating and intersector handshaking, and host microprocessor queueing of the address of the next disc data block to be transferred.

40 Claims, 8 Drawing Sheets



EAST Browser - L5: (23) l4 and (proc... | US 5687389 | Tag: S | Doc: 11/23 | Front Page 1/1 (Total images 40) | Fron...
File
Edit
View
Tools
Window
Help

US005687389A

United States Patent [19]

Packer

[11] **Patent Number: 5,687,389**

[45] **Date of Patent: Nov. 11, 1997**

[54] **SYSTEM FOR CONTROLLING AN AUTOMATIC READ OPERATION OF READ CACHE CONTROL CIRCUIT IN A DISK DRIVE CONTROLLER UTILIZING A START COUNTER, A WORKING COUNTER, AND A SECTOR COUNTER**

[75] Inventor: John S. Packer, Milpitas, Calif.

[73] Assignee: Adaptec, Inc., Milpitas, Calif.

[21] Appl. No.: 173,529

[22] Filed: Dec. 22, 1993

[51] Int. Cl.^{6} G06F 9/26; G06F 12/02; G06F 13/372

[52] U.S. Cl. 395/828; 395/826; 395/853; 395/854; 395/445; 365/50; 369/47; 364/236.2; 364/239.7; 364/243.4; 364/246.7

[58] Field of Search 395/275, 800, 395/425, 853, 825, 826, 445, 854; 365/50; 369/47

[56] **References Cited**

U.S. PATENT DOCUMENTS

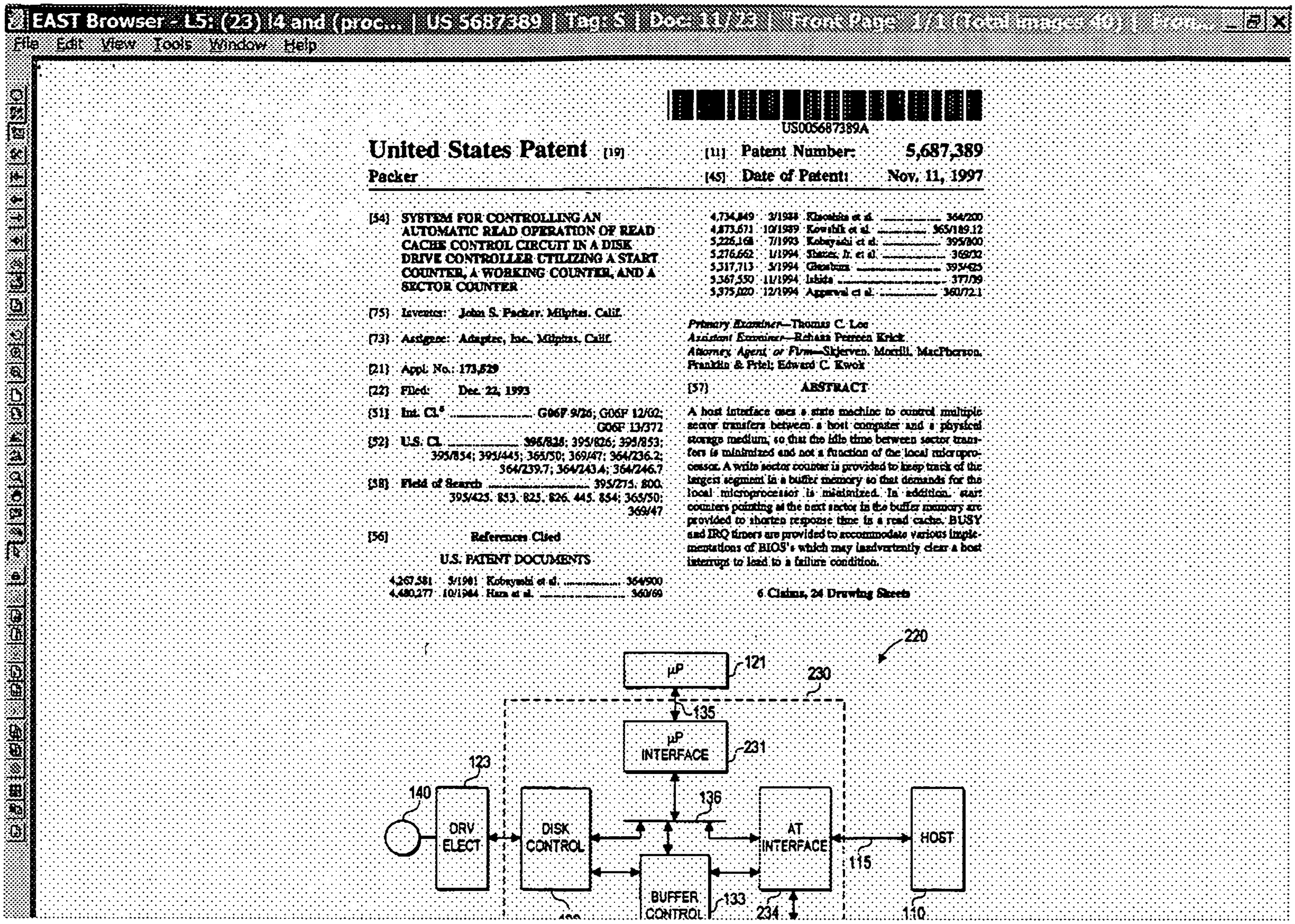
4,267,581	5/1981	Kobayashi et al.	364/900
4,480,277	10/1984	Hara et al.	360/69
4,734,849	3/1988	Kinoshita et al.	364/200
4,873,671	10/1989	Kowalik et al.	365/189.12
5,226,168	7/1993	Kobayashi et al.	395/800
5,276,662	1/1994	Shaver, Jr. et al.	369/32
5,317,713	5/1994	Glassburn	395/425
5,367,550	11/1994	Ishida	377/39
5,375,020	12/1994	Aggarwal et al.	360/72.1

Primary Examiner—Thomas C. Lee
Assistant Examiner—Rehana Perveen Krick
Attorney, Agent, or Firm—Skjerven, Morrill, MacPherson, Franklin & Friel; Edward C. Kwok

[57] **ABSTRACT**

A host interface uses a state machine to control multiple sector transfers between a host computer and a physical storage medium, so that the idle time between sector transfers is minimized and not a function of the local microprocessor. A write sector counter is provided to keep track of the largest segment in a buffer memory so that demands for the local microprocessor is minimized. In addition, start counters pointing at the next sector in the buffer memory are provided to shorten response time in a read cache. BUSY and IRQ timers are provided to accommodate various implementations of BIOS's which may inadvertently clear a host interrupt to lead to a failure condition.

6 Claims, 24 Drawing Sheets



EAST Browser - L5: (23) l4 and (proc... | US 5687389 | Tag: S | Doc: 11/23 | Front Page: 1/1 (Total images 40) | Fron...
File Edit View Tools Window Help

US005687389A

United States Patent [19]

Packer

[11] Patent Number: **5,687,389**

[45] Date of Patent: **Nov. 11, 1997**

[54] **SYSTEM FOR CONTROLLING AN AUTOMATIC READ OPERATION OF READ CACHE CONTROL CIRCUIT IN A DISK DRIVE CONTROLLER UTILIZING A START COUNTER, A WORKING COUNTER, AND A SECTOR COUNTER**

[75] Inventor: John S. Packer, Milpitas, Calif.

[73] Assignee: Adaptec, Inc., Milpitas, Calif.

[21] Appl. No.: 173,529

[22] Filed: Dec. 22, 1993

[51] Int. Cl.^{6} G06F 9/26; G06F 12/02; G06F 13/372

[52] U.S. Cl. 395/825; 395/826; 395/853; 395/854; 395/445; 365/50; 369/47; 364/236.2; 364/239.7; 364/243.4; 364/246.7

[58] Field of Search 395/275, 800, 395/425, 853, 825, 826, 445, 854; 365/50; 369/47

[56] References Cited

U.S. PATENT DOCUMENTS

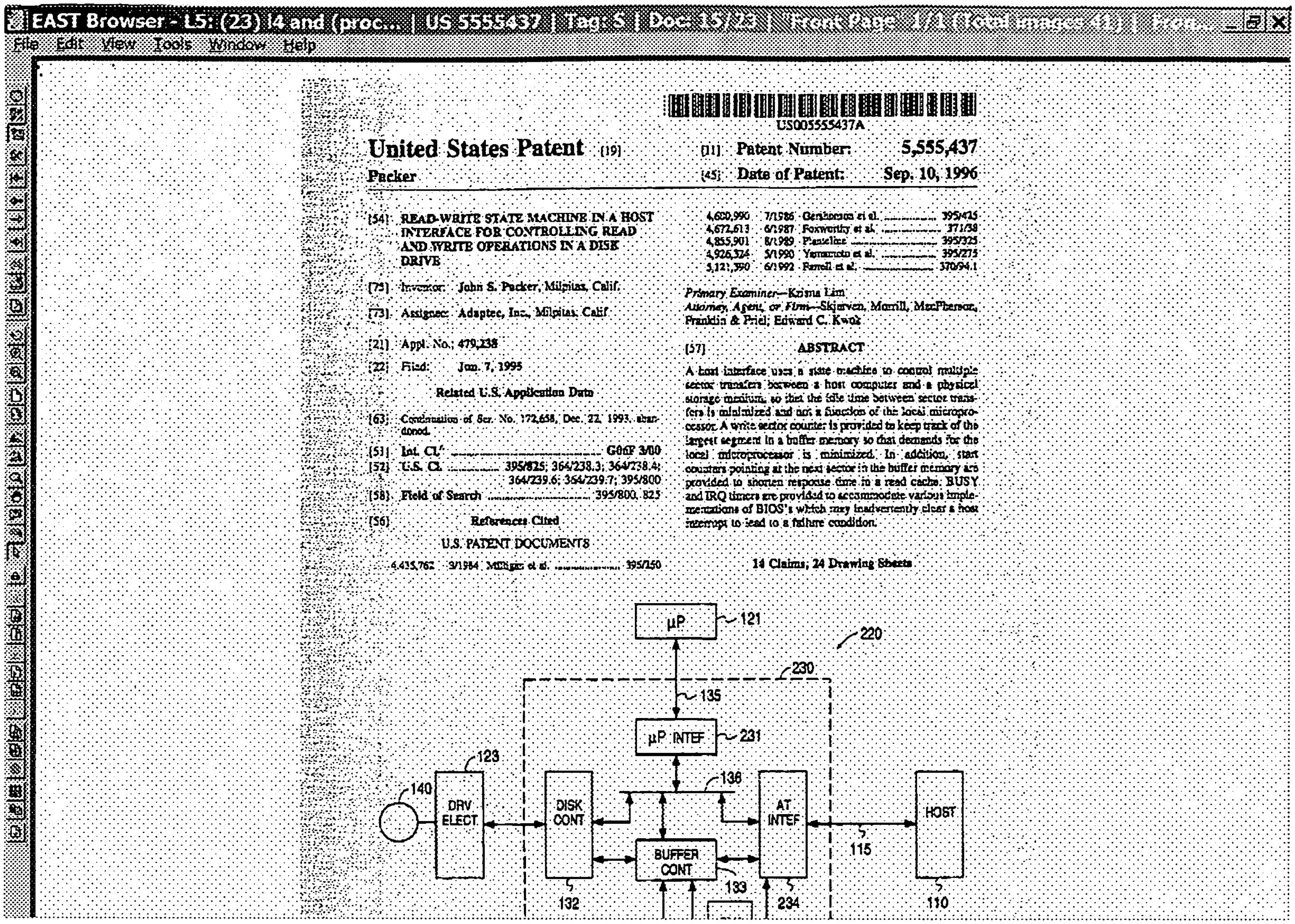
4,267,581	5/1981	Kobayashi et al.	364/900
4,480,277	10/1984	Hara et al.	360/69
4,734,849	3/1988	Kinoshita et al.	364/200
4,873,671	10/1989	Kowshik et al.	365/189.12
5,226,168	7/1993	Kobayashi et al.	395/800
5,276,662	1/1994	Shaver, Jr. et al.	369/32
5,317,713	5/1994	Glassburn	395/425
5,367,550	11/1994	Ishida	377/39
5,375,020	12/1994	Aggarwal et al.	360/72.1

Primary Examiner—Thomas C. Lee
Assistant Examiner—Rehana Perveen Krick
Attorney, Agent, or Firm—Skjerven, Morrill, MacPherson, Franklin & Friel; Edward C. Kwok

[57] ABSTRACT

A host interface uses a state machine to control multiple sector transfers between a host computer and a physical storage medium, so that the idle time between sector transfers is minimized and not a function of the local microprocessor. A write sector counter is provided to keep track of the largest segment in a buffer memory so that demands for the local microprocessor is minimized. In addition, start counters pointing at the next sector in the buffer memory are provided to shorten response time in a read cache. BUSY and IRQ timers are provided to accommodate various implementations of BIOS's which may inadvertently clear a host interrupt to lead to a failure condition.

6 Claims, 24 Drawing Sheets



EAST Browser - L5: (23) l4 and (proc... | US 5555437 | Tag: S | Doc: 15/23 | Front Page 1/1 (Total images 41) | Fron...
File Edit View Tools Window Help

US005555437A

United States Patent [19]

Packer

[11] **Patent Number: 5,555,437**

[45] **Date of Patent: Sep. 10, 1996**

[54] **READ-WRITE STATE MACHINE IN A HOST INTERFACE FOR CONTROLLING READ AND WRITE OPERATIONS IN A DISK DRIVE**

[75] Inventor: **John S. Packer**, Milpitas, Calif.

[73] Assignee: **Adaptec, Inc.**, Milpitas, Calif.

[21] Appl. No.: 479,238

[22] Filed: Jun. 7, 1995

Related U.S. Application Data

[63] Continuation of Ser. No. 172,658, Dec. 22, 1993, abandoned.

[51] Int. Cl.⁶ G06F 3/00

[52] U.S. Cl. 395/825; 364/238.3; 364/238.4; 364/239.6; 364/239.7; 395/800

[58] Field of Search 395/800, 825

[56] **References Cited**

U.S. PATENT DOCUMENTS

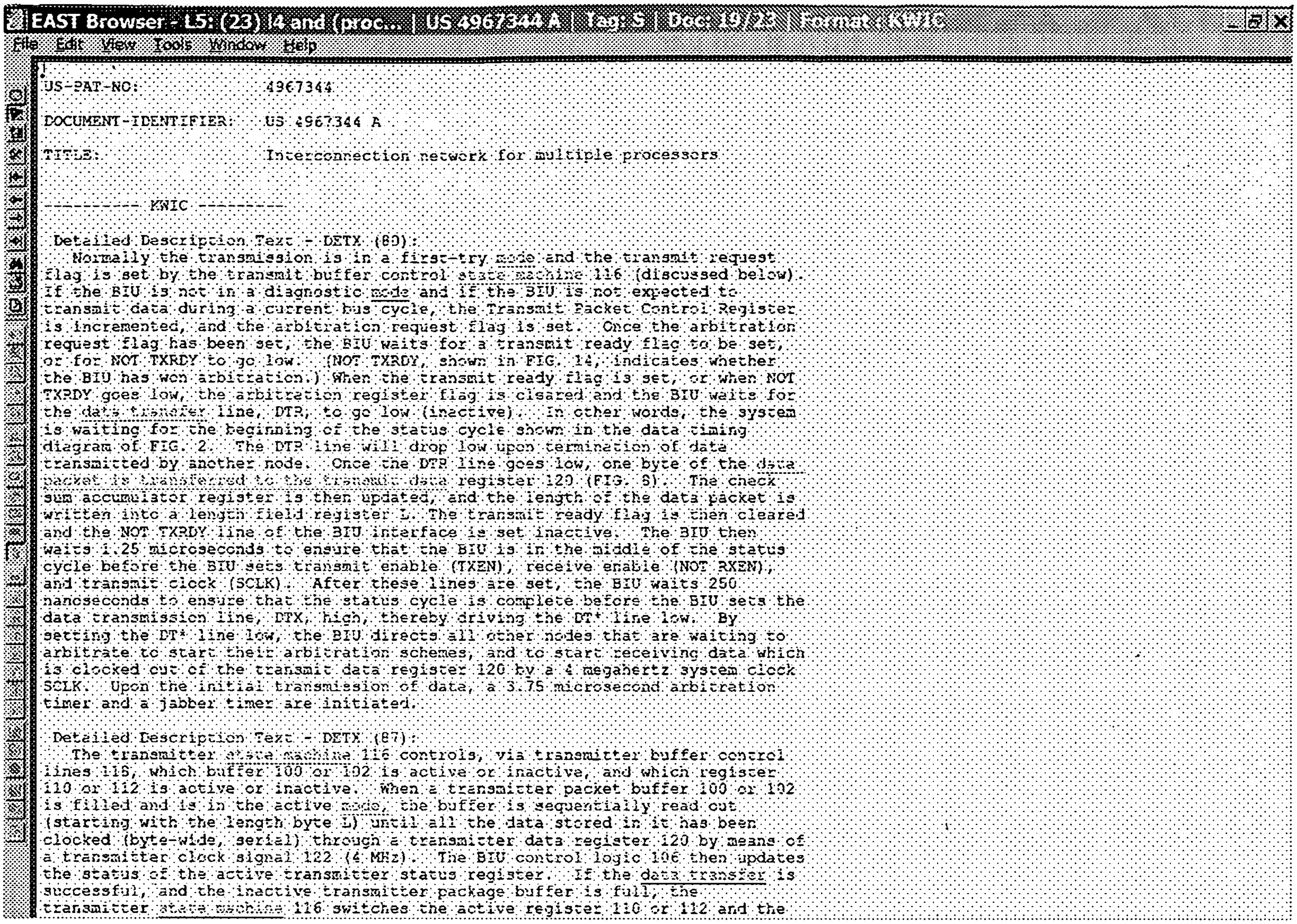
4,435,762	3/1984	Milligan et al.	395/250
4,600,990	7/1986	Gershenson et al.	395/425
4,672,613	6/1987	Foxworthy et al.	371/38
4,855,901	8/1989	Planteline	395/325
4,926,324	5/1990	Yamamoto et al.	395/275
5,121,390	6/1992	Farrell et al.	370/94.1

Primary Examiner—Krisna Lim
Attorney, Agent, or Firm—Skjerven, Morrill, MacPherson, Franklin & Friel; Edward C. Kwok

[57] **ABSTRACT**

A host interface uses a state machine to control multiple sector transfers between a host computer and a physical storage medium, so that the idle time between sector transfers is minimized and not a function of the local microprocessor. A write sector counter is provided to keep track of the largest segment in a buffer memory so that demands for the local microprocessor is minimized. In addition, start counters pointing at the next sector in the buffer memory are provided to shorten response time in a read cache. BUSY and IRQ timers are provided to accommodate various implementations of BIOS's which may inadvertently clear a host interrupt to lead to a failure condition.

14 Claims, 24 Drawing Sheets


EAST Browser - L5: (23) l4 and (proc... | US 4967344 A | Tag: S | Doc: 19/23 | Format : KWIC
File Edit View Tools Window Help

US-PAT-NO: 4967344

DOCUMENT-IDENTIFIER: US 4967344 A

TITLE: Interconnection network for multiple processors

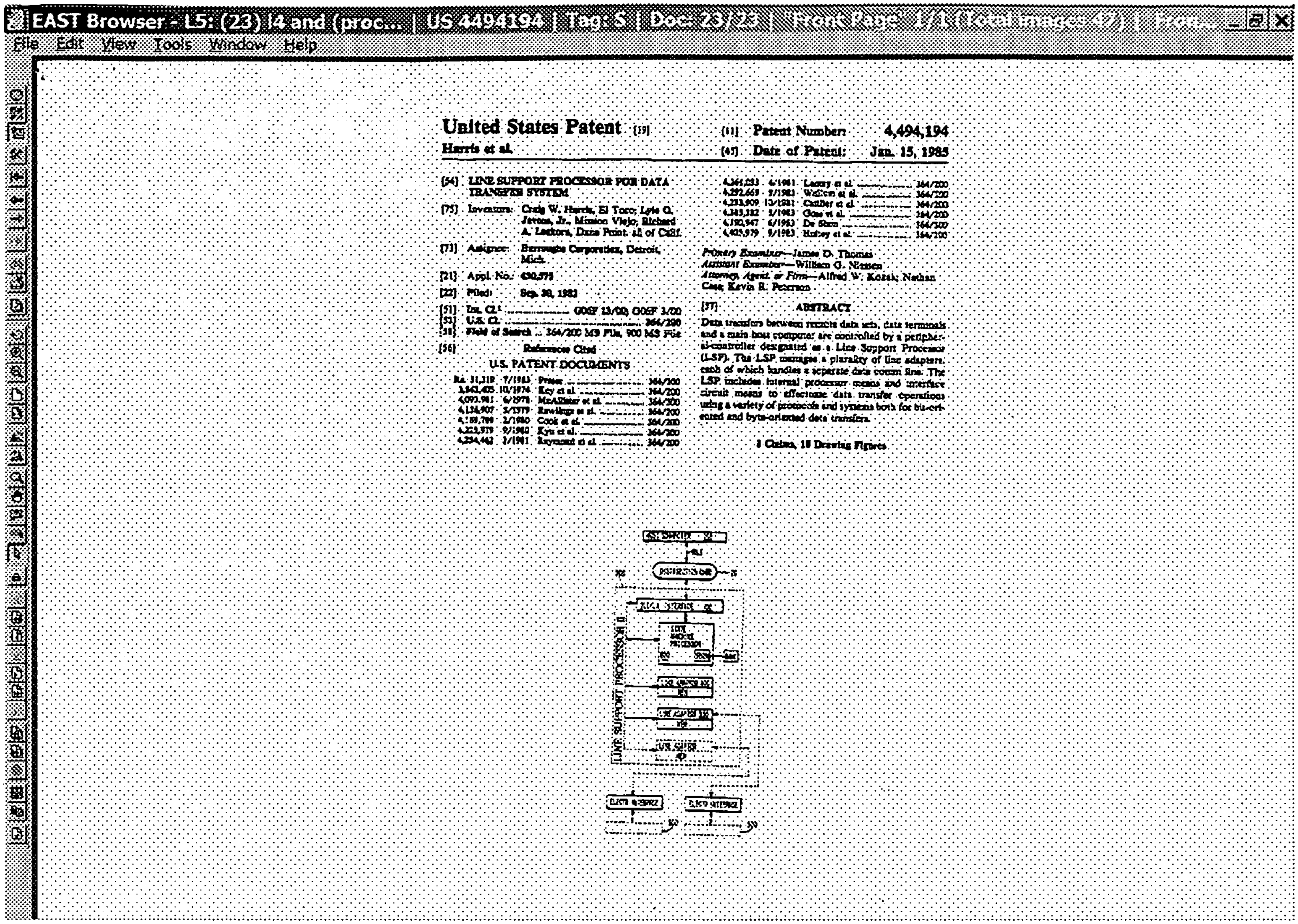
---------- KWIC ---------

Detailed Description Text - DETX (80):

Normally the transmission is in a first-try mode and the transmit request flag is set by the transmit buffer control state machine 116 (discussed below). If the BIU is not in a diagnostic mode and if the BIU is not expected to transmit data during a current bus cycle, the Transmit Packet Control Register is incremented, and the arbitration request flag is set. Once the arbitration request flag has been set, the BIU waits for a transmit ready flag to be set, or for NOT TXRDY to go low. (NOT TXRDY, shown in FIG. 14, indicates whether the BIU has won arbitration.) When the transmit ready flag is set, or when NOT TXRDY goes low, the arbitration register flag is cleared and the BIU waits for the data transfer line, DTR, to go low (inactive). In other words, the system is waiting for the beginning of the status cycle shown in the data timing diagram of FIG. 2. The DTR line will drop low upon termination of data transmitted by another node. Once the DTR line goes low, one byte of the data packet is transferred to the transmit data register 120 (FIG. 8). The check sum accumulator register is then updated, and the length of the data packet is written into a length field register L. The transmit ready flag is then cleared and the NOT TXRDY line of the BIU interface is set inactive. The BIU then waits 1.25 microseconds to ensure that the BIU is in the middle of the status cycle before the BIU sets transmit enable (TXEN), receive enable (NOT RXEN), and transmit clock (SCLK). After these lines are set, the BIU waits 250 nanoseconds to ensure that the status cycle is complete before the BIU sets the data transmission line, DTX, high, thereby driving the DT* line low. By setting the DT* line low, the BIU directs all other nodes that are waiting to arbitrate to start their arbitration schemes, and to start receiving data which is clocked out of the transmit data register 120 by a 4 megahertz system clock SCLK. Upon the initial transmission of data, a 3.75 microsecond arbitration timer and a jabber timer are initiated.

Detailed Description Text - DETX (87):

The transmitter state machine 116 controls, via transmitter buffer control lines 118, which buffer 100 or 102 is active or inactive, and which register 110 or 112 is active or inactive. When a transmitter packet buffer 100 or 102 is filled and is in the active mode, the buffer is sequentially read out (starting with the length byte L) until all the data stored in it has been clocked (byte-wide, serial) through a transmitter data register 120 by means of a transmitter clock signal 122 (4 MHz). The BIU control logic 106 then updates the status of the active transmitter status register. If the data transfer is successful, and the inactive transmitter package buffer is full, the transmitter state machine 116 switches the active register 110 or 112 and the



EAST Browser - L5: (23) l4 and (proc... | US 4494194 | Tag: S | Doc: 23/23 | Front Page: 1/1 (Total images: 47) | Fron...

United States Patent [19]

Harris et al.

[11] Patent Number: 4,494,194

[45] Date of Patent: Jan. 15, 1985

[54] LINE SUPPORT PROCESSOR FOR DATA TRANSFER SYSTEM

[75] Inventors: Craig W. Harris, El Toro; Lyle O. Jevons, Jr., Mission Viejo; Richard A. Lesbora, Dana Point, all of Calif.

[73] Assignee: Burroughs Corporation, Detroit, Mich.

[21] Appl. No.: 430,575

[22] Filed: Sep. 30, 1982

[51] Int. Cl.³ G06F 13/00; G06F 3/00

[52] U.S. Cl. 364/200

[58] Field of Search ... 364/200 MS File, 900 MS File

[56] References Cited

U.S. PATENT DOCUMENTS

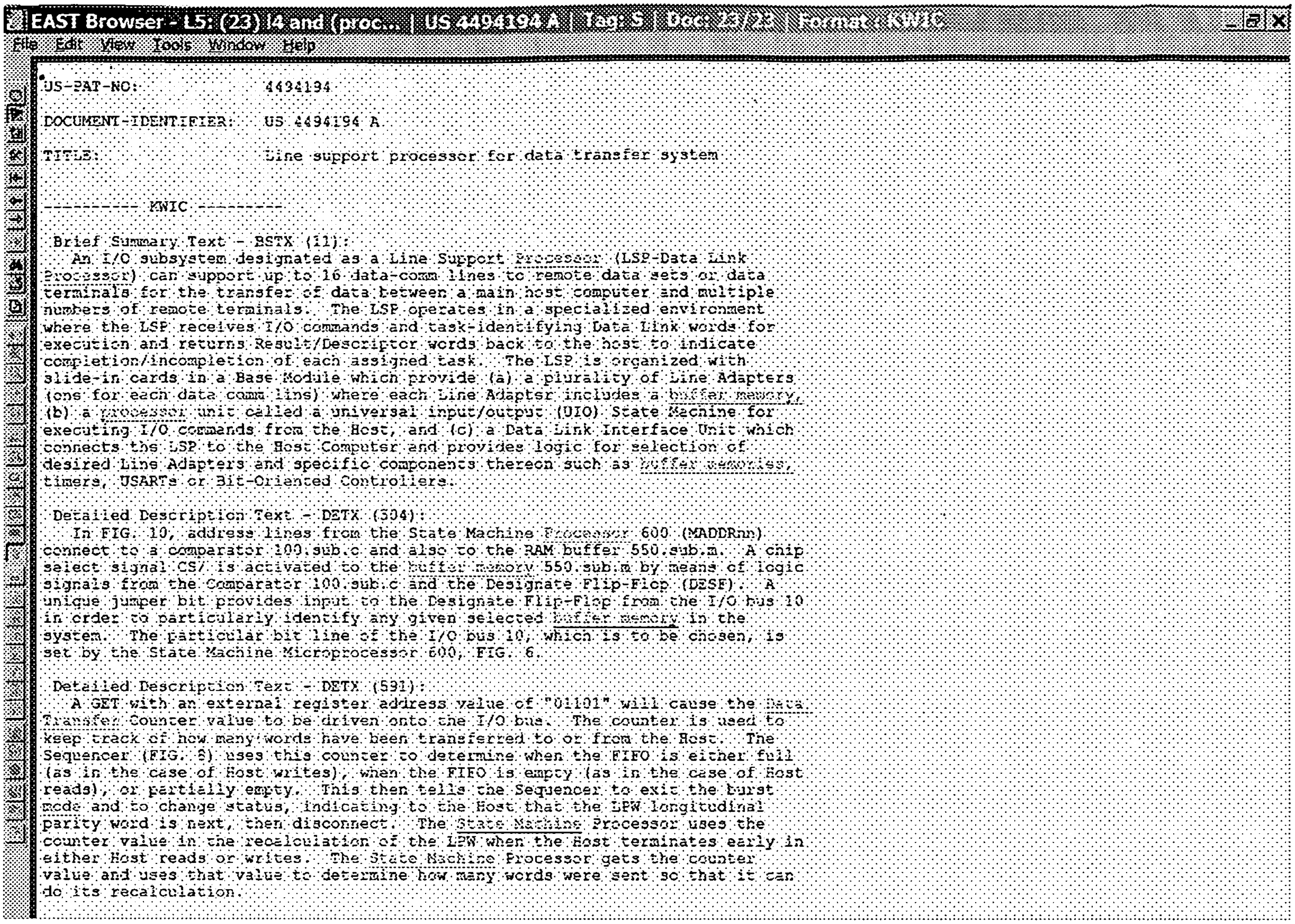
Re. 31,319	7/1983	Prince	364/200
3,842,405	10/1974	Key et al.	364/200
4,093,981	6/1978	McAllister et al.	364/200
4,134,907	5/1979	Rawlings et al.	364/200
4,189,769	2/1980	Cook et al.	364/200
4,225,919	9/1980	Kyu et al.	364/200
4,254,462	3/1981	Raymond et al.	364/200
4,261,033	4/1981	Leary et al.	364/200
4,292,669	9/1981	Wollum et al.	364/200
4,293,909	10/1981	Catiller et al.	364/200
4,383,312	5/1983	Goss et al.	364/200
4,392,947	6/1983	De Shon	364/200
4,403,979	9/1983	Hatley et al.	364/200

Primary Examiner—James D. Thomas
Assistant Examiner—William G. Niessen
Attorney, Agent, or Firm—Alfred W. Kozak; Nathan Cass; Kevin R. Peterson

[57] ABSTRACT

Data transfers between remote data sets, data terminals and a main host computer are controlled by a peripheral-controller designated as a Line Support Processor (LSP). The LSP manages a plurality of line adapters, each of which handles a separate data comm line. The LSP includes internal processor means and interface circuit means to effectuate data transfer operations using a variety of protocols and systems both for bit-oriented and byte-oriented data transfers.

1 Claim, 18 Drawing Figures


EAST Browser - L5: (23) l4 and (proc... | US 4494194 A | Tag: S | Doc: 23/23 | Format : KWIC
File Edit View Tools Window Help

US-PAT-NO: 4494194

DOCUMENT-IDENTIFIER: US 4494194 A

TITLE: Line support processor for data transfer system

---------- KWIC ----------

Brief Summary Text - BSTX (11):

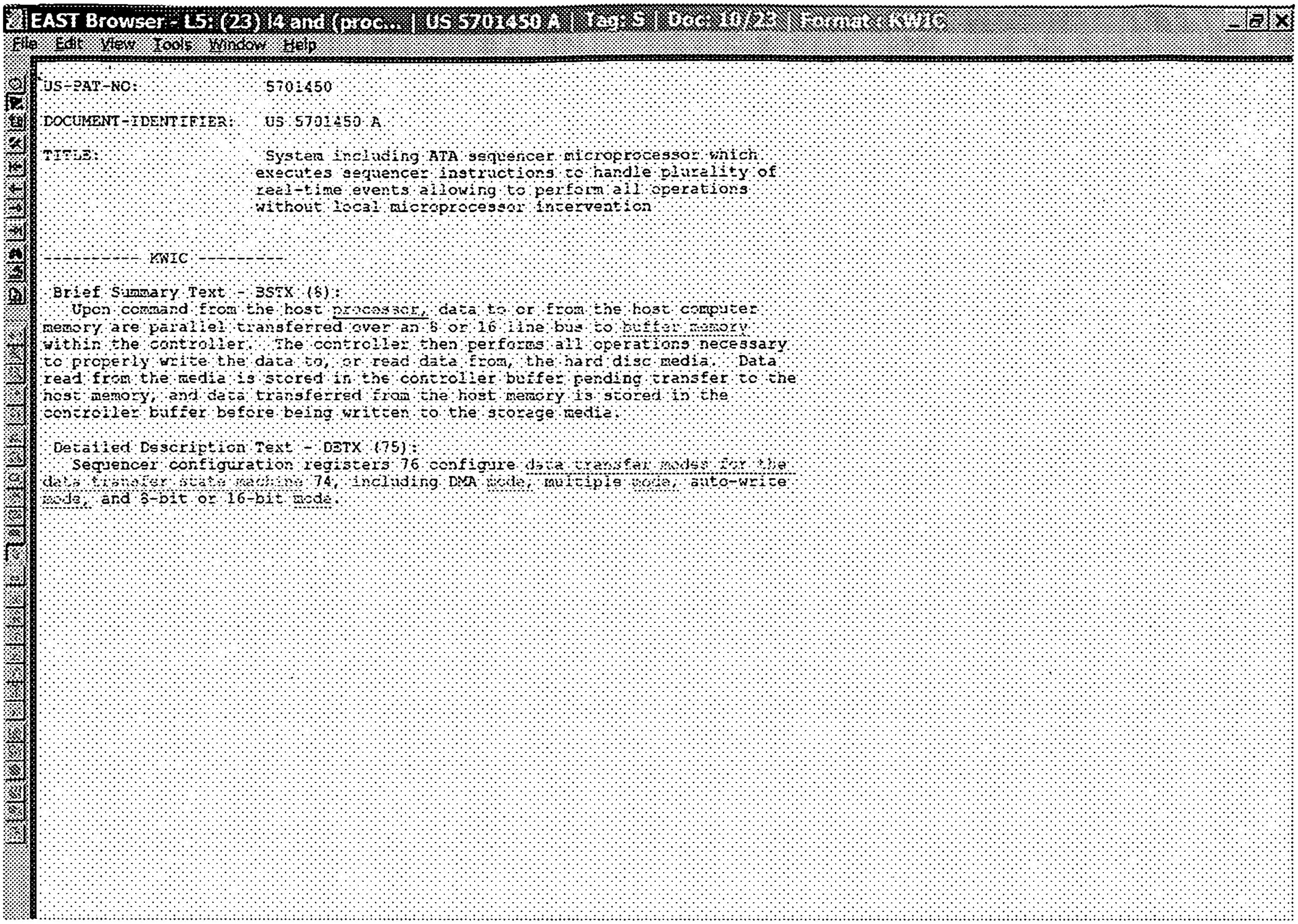
An I/O subsystem designated as a Line Support Processor (LSP-Data Link Processor) can support up to 16 data-comm lines to remote data sets or data terminals for the transfer of data between a main host computer and multiple numbers of remote terminals. The LSP operates in a specialized environment where the LSP receives I/O commands and task-identifying Data Link words for execution and returns Result/Descriptor words back to the host to indicate completion/incompletion of each assigned task. The LSP is organized with slide-in cards in a Base Module which provide (a) a plurality of Line Adapters (one for each data comm line) where each Line Adapter includes a buffer memory, (b) a processor unit called a universal input/output (UIO) State Machine for executing I/O commands from the Host, and (c) a Data Link Interface Unit which connects the LSP to the Host Computer and provides logic for selection of desired Line Adapters and specific components thereon such as buffer memories, timers, USARTs or Bit-Oriented Controllers.

Detailed Description Text - DETX (304):

In FIG. 10, address lines from the State Machine Processor 600 (MADDRnn) connect to a comparator 100.sub.c and also to the RAM buffer 550.sub.m. A chip select signal CS/ is activated to the buffer memory 550.sub.m by means of logic signals from the Comparator 100.sub.c and the Designate Flip-Flop (DESF). A unique jumper bit provides input to the Designate Flip-Flop from the I/O bus 10 in order to particularly identify any given selected buffer memory in the system. The particular bit line of the I/O bus 10, which is to be chosen, is set by the State Machine Microprocessor 600, FIG. 6.

Detailed Description Text - DETX (591):

A GET with an external register address value of "01101" will cause the Data Transfer Counter value to be driven onto the I/O bus. The counter is used to keep track of how many words have been transferred to or from the Host. The Sequencer (FIG. 8) uses this counter to determine when the FIFO is either full (as in the case of Host writes), when the FIFO is empty (as in the case of Host reads), or partially empty. This then tells the Sequencer to exit the burst mode and to change status, indicating to the Host that the LPW longitudinal parity word is next, then disconnect. The State Machine Processor uses the counter value in the recalculation of the LPW when the Host terminates early in either Host reads or writes. The State Machine Processor gets the counter value and uses that value to determine how many words were sent so that it can do its recalculation.


EAST Browser - L5: (23) l4 and (proc... | US 5701450 A | Tag: S | Doc: 10/23 | Format : KWIC
File
Edit
View
Tools
Window
Help

US-PAT-NO: 5701450

DOCUMENT-IDENTIFIER: US 5701450 A

TITLE: System including ATA sequencer microprocessor which executes sequencer instructions to handle plurality of real-time events allowing to perform all operations without local microprocessor intervention

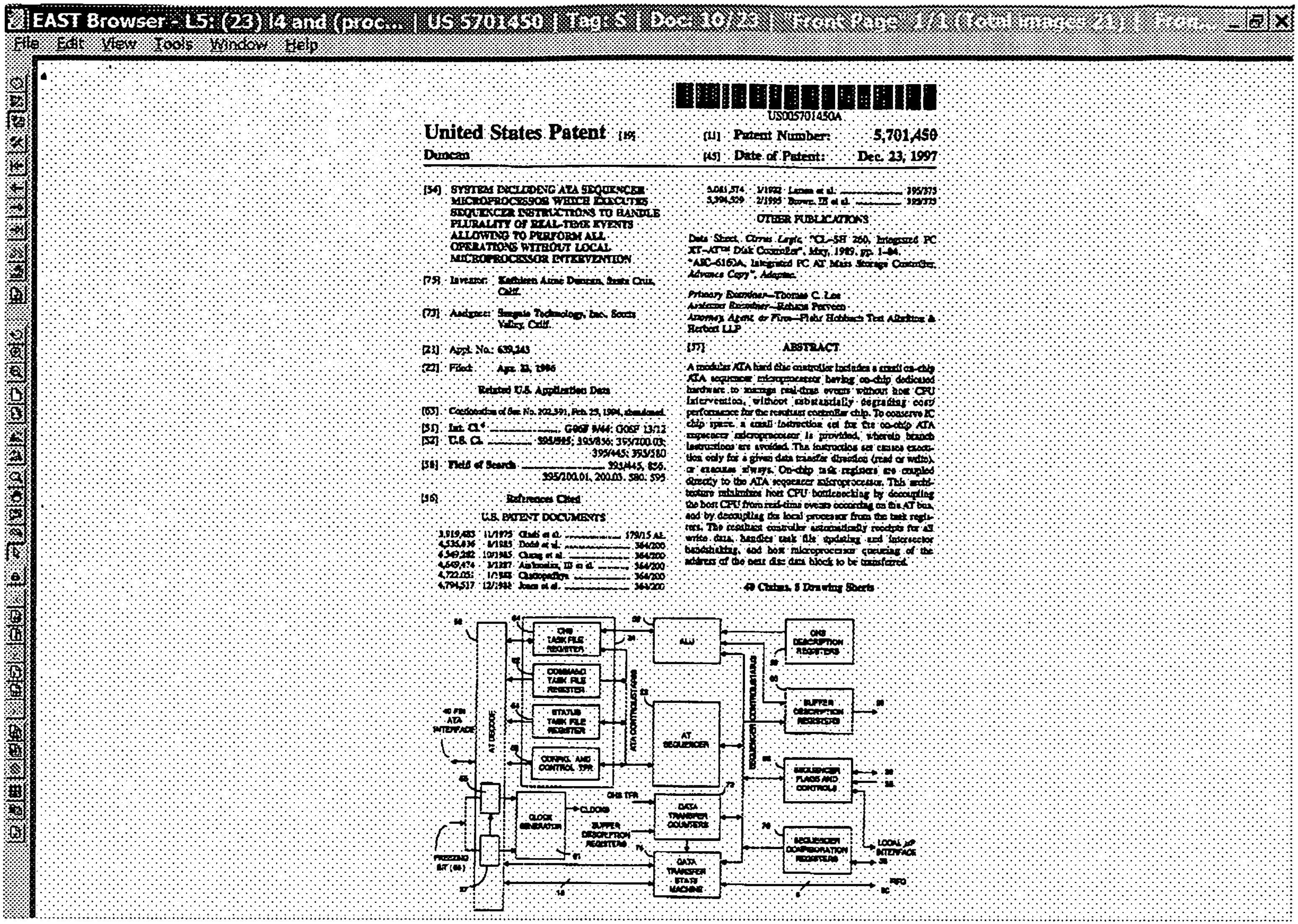
---------- KWIC ---------

Brief Summary Text - BSTX (8):

Upon command from the host processor, data to or from the host computer memory are parallel transferred over an 8 or 16 line bus to buffer memory within the controller. The controller then performs all operations necessary to properly write the data to, or read data from, the hard disc media. Data read from the media is stored in the controller buffer pending transfer to the host memory, and data transferred from the host memory is stored in the controller buffer before being written to the storage media.

Detailed Description Text - DETX (75):

Sequencer configuration registers 76 configure data transfer modes for the data transfer state machine 74, including DMA mode, multiple mode, auto-write mode, and 8-bit or 16-bit mode.


EAST Browser - L5: (23) l4 and (proc... | US 5701450 | Tag: S | Doc: 10/23 | "Front Page" 1/1 (Total images: 21) | From...
File
Edit
View
Tools
Window
Help

US005701450A

United States Patent [19]

Duncan

[11] Patent Number: 5,701,450

[45] Date of Patent: Dec. 23, 1997

[54] SYSTEM INCLUDING ATA SEQUENCER MICROPROCESSOR WHICH EXECUTES SEQUENCER INSTRUCTIONS TO HANDLE PLURALITY OF REAL-TIME EVENTS ALLOWING TO PERFORM ALL OPERATIONS WITHOUT LOCAL MICROPROCESSOR INTERVENTION

[75] Inventor: Kathleen Anne Duncan, Santa Cruz, Calif.

[73] Assignee: Seagate Technology, Inc., Scotts Valley, Calif.

[21] Appl. No.: 639,243

[22] Filed: Apr. 23, 1996

Related U.S. Application Data

[63] Continuation of Ser. No. 202,391, Feb. 25, 1994, abandoned.

[51] Int. Cl.⁶ G06F 9/44; G06F 13/12

[52] U.S. Cl. 395/585; 395/856; 395/200.03; 395/445; 395/580

[58] Field of Search 395/445, 856, 395/200.01, 200.03, 580, 595

[56] References Cited

U.S. PATENT DOCUMENTS

3,919,483	11/1975	Clark et al.	179/15 AL
4,535,836	8/1985	Dodd et al.	364/200
4,549,282	10/1985	Chang et al.	364/200
4,649,474	3/1987	Ambrosius, III et al.	364/200
4,722,051	1/1988	Chattopadhya	364/200
4,794,517	12/1988	Jones et al.	364/200
5,081,574	1/1992	Larsen et al.	395/375
5,394,529	2/1995	Brown, III et al.	395/375

OTHER PUBLICATIONS

Data Sheet, *Cirrus Logic*, "CL-SH 260, Integrated PC XT-AT™ Disk Controller", May, 1989, pp. 1-84.

"AIC-6160A, Integrated PC AT Mass Storage Controller, *Advance Copy*", *Adaptec*.

Primary Examiner—Thomas C. Lee
Assistant Examiner—Rehana Perveen
Attorney, Agent, or Firm—Flehr Hohbach Test Albritton & Herbert LLP

[57] ABSTRACT

A modular ATA hard disc controller includes a small on-chip ATA sequencer microprocessor having on-chip dedicated hardware to manage real-time events without host CPU intervention, without substantially degrading cost/performance for the resultant controller chip. To conserve IC chip space, a small instruction set for the on-chip ATA sequencer microprocessor is provided, wherein branch instructions are avoided. The instruction set causes execution only for a given data transfer direction (read or write), or executes always. On-chip task registers are coupled directly to the ATA sequencer microprocessor. This architecture minimizes host CPU bottlenecking by decoupling the host CPU from real-time events occurring on the AT bus, and by decoupling the local processor from the task registers. The resultant controller automatically receipts for all write data, handles task file updating and intersector handshaking, and host microprocessor queuing of the address of the next disc data block to be transferred.

49 Claims, 5 Drawing Sheets